

Software Engineering 1

Abgabedokument

Teilaufgabe 1

(Anforderungsanalyse und Planungsphase)

Persönliche Daten, bitte vollständig ausfüllen:

Nachname, Vorname:	Jahic Enan
Matrikelnummer:	11708481
E-Mail-Adresse:	A11708481@unet.univie.ac.at
Datum:	28.03.2021

Aufgabe 1: Anforderungsanalyse (2 Punkte)

Analysieren der Spielidee und des Netzwerkprotokolls um 8 Anforderungen (bestehend zumindest aus 3 funktionalen, 3 nichtfunktionalen und einer Designbedingung) nach den folgenden Kriterien zu dokumentieren. Achten Sie darauf den im Skriptum und der Vorlesung behandelten Qualitätsaspekten Genüge zu tun.

Typ der Anforderung: funktional

Anforderung 1

- **Beschreibung:** Generierung der gesamten Karte-jede KI erstellt seine eigene Hälfte der Karte
- **Bezugsquelle:** [Spielidee]-Die Karte, auf welcher gespielt wird, ist hierbei nicht fest vorgegeben, sondern wird von beiden KIs selbstständig und automatisch beim Start des Spiels erstellt (jeweils eine Hälfte der Karte).

Anforderung 2

Beschreibung: Schatzaufnahme - der Spieler kann nur einen Schatz aufnehmen, der sich auf seiner Kartenhälfte befindet.

- **Bezugsquelle:** [Spielidee], Ein Schatz kann nur von der Spielfigur der zugeordneten KI "aufgenommen" und "verwendet" werden.

Anforderung 3

Beschreibung: Vereinigung von Kartenhälften-Server wird die Kartenhälften vereinigen, nachdem die beiden Spieler gültige Karten an den Server gesendet haben.

)

- **Bezugsquelle:** [Spielidee], „Hierzu erstellt jede der beiden KIs zufällig eine Hälfte der finalen Spielkarte (mit je 4 x 8 Feldern für eine endgültige Kartengröße von 8 x 8 oder 4 x 16). Zu beachten ist hierbei, dass die Kartenhälften entweder an den kurzen (Kartenmaße 4 x 16) oder den langen Seiten (Kartenmaße 8 x 8) kombiniert werden – dies kann der Client nicht beeinflussen, sondern wird vom Server durchgeführt.“

Anforderung 4

- **Beschreibung:** Wasser ist tödlich-die KI muss vermeiden, dass die Figur ins Wasser geht.
- **Bezugsquelle:** Spielidee – „Außerdem darf **Wasser unter keinen Umständen betreten** werden. Bewegt sich die Spielfigur einer KI in ein Wasserfeld, verliert die KI automatisch.“

Typ der Anforderung: nicht funktional

Anforderung 5

- **Beschreibung:** Maximale Spieleranzahl von zwei- Der Server stellt sicher, dass sich maximal zwei Spieler in einem Spiel befinden.
- **Bezugsquelle:** Spielidee, „Die grundlegende Spielidee ist, dass zwei KIs auf der gleichen Spielkarte eine vergleichbare Aufgabe erfüllen müssen. Die KI, die diese schneller erfüllt, gewinnt das Spiel;“

Anforderung 6

- **Beschreibung:** der Server gewährleistet, dass eine KI seine nächste Bewegung in spätestens 3 Sekunden senden kann.
- **Bezugsquelle:** Spielidee, „Um die Spiele für die Zuschauer spannend zu gestalten, wurde festgelegt, dass ein Spiel insgesamt nicht länger als **200 Spielaktionen** dauern darf und eine KI für **jede dieser rundenbasierten Spielaktion nicht mehr als 3** .“

Anforderung 7

- **Beschreibung:** Größe der Karte-der Server stellt absolut sicher, dass die gesamte Karte aus 64 Felder besteht.
- **Bezugsquelle:** Spielidee – „Die gesamte Karte hat immer eine fixe Ausdehnung von 64 Feldern“

Anforderung 8

- **Beschreibung:** Anzahl der Bezüge -der Server stellt absolut sicher, dass in einem Spiel in höchstens 200 Bezügen ein Gewinner gefunden wird.
- **Bezugsquelle:** Spielidee, „Um die Spiele für die Zuschauer spannend zu gestalten, wurde festgelegt, dass ein Spiel insgesamt nicht länger als **200 Spielaktionen** dauern darf und eine KI für **jede dieser rundenbasierten Spielaktion nicht mehr als 3 Sekunden** Bedenkzeit zur Berechnung erhält (relevant sind hierbei Spielerbewegung und Kartengenerierung). Sollten diese Bedingungen nicht erfüllt werden, verliert die KI, welche gerade an der Reihe ist, das Spiel automatisch und der zugehörige menschliche Spieler bekommt dies mitgeteilt.“

Typ der Anforderung: Designbedingung

Anforderung 9

- **Beschreibung:** Kommunikations-Server und Client kommunizieren miteinander über das http Protokoll (get, post Nachrichten).
- **Bezugsquelle:** Netzwerkprotokoll,
- „Die Technologische Basis des Nachrichtenaustauschs stellt eine Restschnittstelle dar, daher es wird das HTTP Protokoll verwendet sowie die zugehörigen Operationen GET und POST.“

Aufgabe 2: Anforderungsdokumentation (2 Punkte)

Dokumentation von *einer* in Aufgabe 1 identifizierten Anforderungen nach dem vorgegebenen Schema. Ziehen Sie eine Anforderung heran, für die alle Bestandteile der Vorlage mit relevantem Inhalt befüllt werden können. Wir empfehlen hierzu eine **funktionale** Anforderung auszuwählen.

Dokumentation Anforderung

- **Name:** Generierung der gesamten Karte
- **Beschreibung und Priorität:** jede KI erstellt seine eigenen Kartenhälfte. Sobald KI die Karte auf Designbedingungen überprüft hat und der Server sagt, dass die KI dran ist, kann sie die dann an den Server schicken.
Priorität: Hoch

- **Relevante Anforderungen:**

Funktionale Anforderungen:

-Vereinigung von Kartenhälften

Nichtfunktionale Anforderungen:

- Größe der Karte

- **Relevante Business Rules:**

1. Jede Kartenhälfte muss mindestens 3 Bergfelder, 15 Wiesenfelder und 4 Wasserfelder beinhalten.
2. Es dürfen keine Inseln generiert werden. Das bedeutet, dass die Bewegung der Figur nicht von Wasserfeldern verhindert werden darf.
3. Bereits zum Beginn des Spieles sind nur die Terrains der Karte beiden KIs vollständig bekannt.
4. Jede der beiden KI's erstellt seine eigene Kartenhälfte.
5. Eine Kartenhälfte muss aus 32 Feldern bestehen.
6. Jede Kartenhälfte ist im 4x8 Format.

7. Die beiden Kartenhälften kann der Server entweder an den langen (8x8 Feldern) oder an den kurzen (4x16 Feldern) Seiten kombinieren.

Impuls/Ergebnis - Typisches Szenario:

Vorbedingungen:

- Ein Mensch hat eine Spiel-ID auf dem Server erzeugt
- Zwei KI's sind mit dem Server mit derselben Spiel-ID erfolgreich verbunden

Hauptsächlicher Ablauf:

- o Die KI erstellt seine Kartenhälfte
- o Die erstellte Kartenhälfte wird von KI überprüft
- o Die erstellte Kartenhälfte hat die Überprüfung bestanden
- o Die KI fragt den Server ob sie an der Reihe ist
- o Wenn der Server mit „Ja“ antwortet, dann sendet die KI die Kartenhälfte
- o Der Server sendet der KI eine responseEnvelope mit dem **status OKAY**.
- o Die zweite KI erstellt ihre Kartenhälfte
- o Die erstellte Kartenhälfte wird von KI überprüft
- o Die erstellte Kartenhälfte hat die Überprüfung bestanden
- o KI fragt den Server ob sie an der Reihe ist
- o Wenn der Server mit „Ja“ antwortet, dann sendet die KI ihre Kartenhälfte
- o Der Server sendet der KI eine responseEnvelope mit dem **status OKAY**.

Nachbedingungen:

- Die Spieler fragen den Server nach dem Spielstatus ab
- Server schickt die ganze Karte zu den Spielern
- Die Spieler fragen den Server nach dem Spielstatus ab, wenn die KI laut dem Spielstatus dran ist kann die ihre Bewegung schicken
- Ein Spieler hat das Spiel gewonnen

- **Impuls/Ergebnis - Alternativszenario:** KI hat fehlerhafte Karte erstellt und muss die noch einmal generieren.

Vorbedingungen:

- o Ein Mensch hat die Spiel-ID auf dem Server erzeugt
- o Zwei KI's sind mit dem Server verbunden worden

Hauptsächlicher Ablauf:

- o Die KI hat ihre Kartenhälfte generiert
- o Die Kartenhälfte wird von der KI überprüft
- o Die Kartenhälfte hat den Test nicht bestanden

Nachbedingungen: Zweifälle:

Fall 1:

- o KI erstellt eine neue Kartenhälfte
- o Die neue Kartenhälfte wird von der KI überprüft
- o Karte hat den Test bestanden
- o KI fragt den Game-state ab und dann weiß dann ob sie an der Reihe ist
- o Server schickt den Game-state
- o Klient schickt die neue Kartenhälfte dem Server

Fall 2:

- Keine gültige Karte wurde innerhalb von 3 Sekunden generiert
- Die KI verliert automatisch das Spiel

- **Impuls/Ergebnis - Fehlerfall:** [Erwarteten Fehlerfall beschreiben und wie dieser gehandhabt wird. Dies ist besonders relevant, da ansonsten nur Gutfälle berücksichtigt werden was im Produktiveinsatz schnell nicht mehr ausreichend ist.]

Vorbedingungen:

- [Ein Mensch hat eine Spiel-ID auf dem Server erzeugt
- Zwei KI's sind mit dem Server mit derselben Spiel-ID erfolgreich verbunden

Hauptsächlicher Ablauf:

- o Die KI erstellt seine Kartenhälfte
- o Die erstellte Kartenhälfte wird von KI überprüft
- o KI fragt den Server ob die die Kartenhälfte schicken darf
- o Wenn der Server mit „Ja“ antwortet dann die KI sendet die Kartenhälfte
- o Der Server überprüft die Kartenhälfte auf Business Rules
- o Karte hat die Regeln nicht bestanden
- o Der Server schickt die **responseEnvelope** mit **exceptionName** , **exceptionMessage** und mit dem **state Error**.

- o **Nachbedingungen:**

Die KI verliert das Spiel

- **Benutzergeschichten:**

Als KI möchte ich, dass die Karte, die ich als erste generiere alle Design-bedienungen erfüllt, um die neue Kartenhälfte nicht generieren zu müssen.

- **Benutzerschnittstelle:**

Ganze Karte wird über CLI dargestellt werden.

Die Karte in CLI kann so dargestellt werden:

```
=====
W W W G G G G G
B G B G G G G A
B W G G G G G G
B G G S G G G G
=====
W W W G G G G G
W G B G G G G A
B W G G G G G G
B G G S G G G G
=====
W -Wasser
G-Grass
S-Schatz
A-Burg
B-Berg
```

Externe Schnittstellen:

Das http Protokoll mit den GET und POST Operationen für den Austausch von Nachrichten zwischen Client und Server. Die ausgetauschten-Daten sind im XML-Format definiert.

Die Nachrichten sind im XML-Format definiert.

Die technologische Basis des Nachrichtenaustauschs stellt eine Restschnittstelle dar, daher es wird das HTTP Protokoll verwendet sowie die zugehörigen Operationen GET und POST. Die ausgetauschten Daten bzw. Nachrichten werden im XML Format definiert bzw. erwartet.

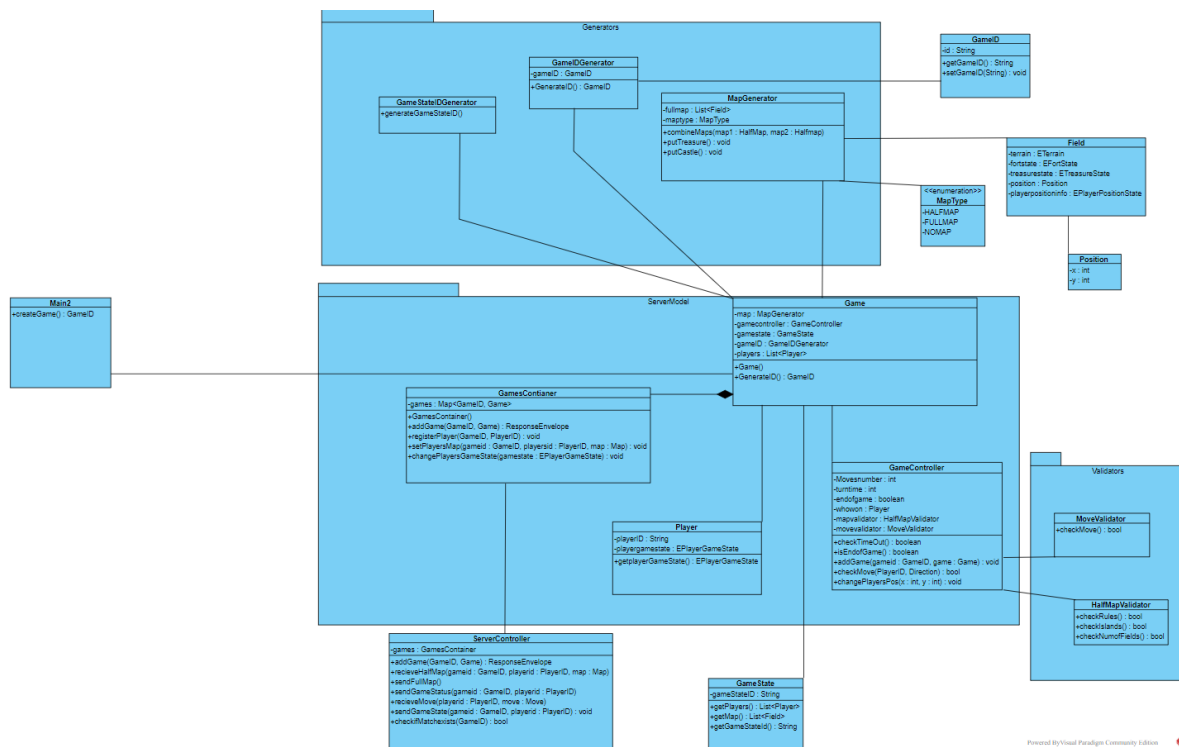
Modellieren Sie händisch alle notwendigen Packages, Klassen und deren Methoden (samt Beziehungen) als zwei UML Klassendiagramme. Achten Sie darauf, dass die Modelle sinnvoll benannte Packages, Klassen, Methoden und Felder beinhalten und die Vorgaben der Spielidee bzw. des Netzwerkprotokolls vollständig in sinnvoller Granularität abgedeckt werden.

TIPP: Beachten Sie zur Ausarbeitung das auf Moodle zu Verfügung gestellte auszugsweise abgebildete 4+1 Diagrammbeispiel. Dieses sollte als Vorlage dienen, und verdeutlicht welche Erwartungen an das Klassendiagramm beziehungsweise die dazugehörigen Sequenzdiagramme gestellt werden (Darstellung, Inhalte, Detailgrad, etc.) und wie diese zusammenhängen.

Alle Diagramme sind in einem Separaten File hochgeladen.

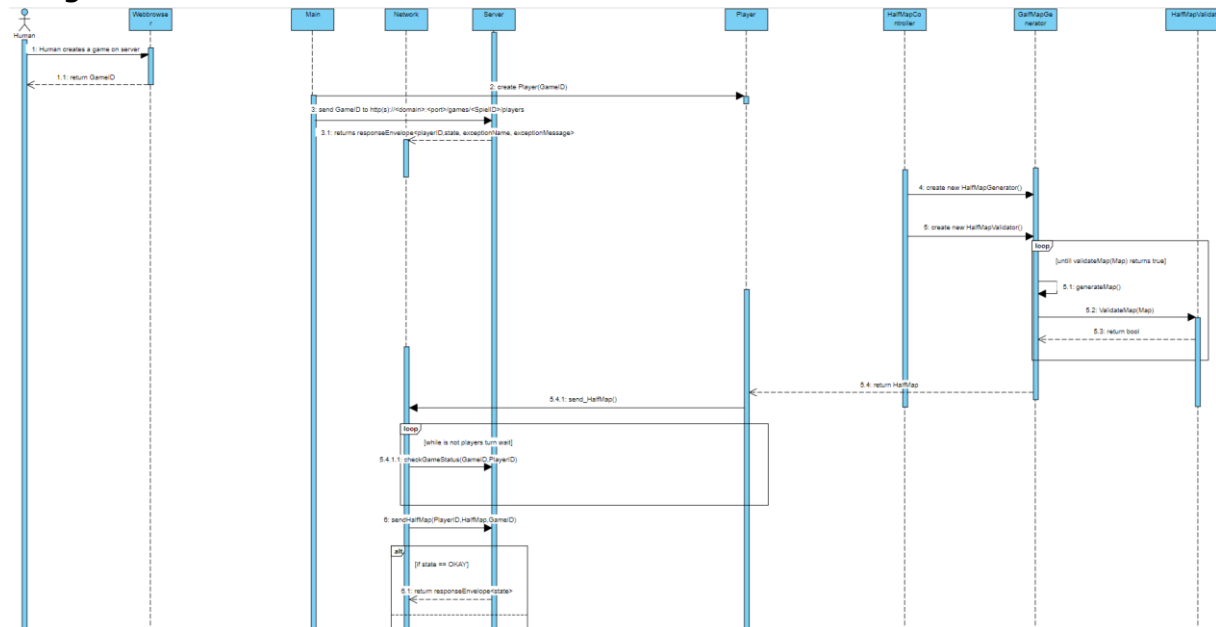
[illegible]

2)Server Class Diagram



3)Client Sequence

Diagramm



4)Server Sequence Diagramm

